UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/712,384	11/12/2003	William John Gallagher	ORACL-01316US3	9603

80548   7590   08/27/2008
Fliesler Meyer LLP
650 California Street
14th Floor
San Francisco, CA 94108

EXAMINER
NGUYEN, PHILLIP H

ART UNIT	PAPER NUMBER
2191	

MAIL DATE	DELIVERY MODE
08/27/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

PTOL-90A (Rev. 04/07)

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>*17 June 2008*</u>.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) <u>*1,10-13,16-18 and 25-27*</u> is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) <u>*1,10-13,16-18 and 25-27*</u> is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date <u>*06172008*</u>.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____ .

DETAILED ACTION

1.   This action is in response to the amendment filed 6/17/2008.

2.   Per applicant's request, claim 1 is amended; claims 1, 10-13, 16-18, and 25-27

remain pending and have been considered below.

Response to Amendment

3.   In view of applicant's amendment, the prior rejection to claim 1 under 112,

second paragraph is hereby withdrawn.

Response to Arguments

4.   Applicant's arguments filed 6/17/2008 have been fully considered but they are

not deemed persuasive.

        Applicant asserts on page 7 of the amendment that Bentley fails to teach

"**automatically** creating a class file container object" and "**dynamically** code

generation."

        Examiner respectfully disagrees with the allegations as argued. Dynamically

generating code means that the code is dynamically (i.e. **at runtime**) created. Bentley

teaches "*The **dynamic framework executes on the platform** and interfaces to the

kernel, provides a platform-independent visual interface between the CMS and a CMS*

user, and employs the services of the kernel" (see at least col. 6:8-12). Bentley goes on

to describe "*As was discussed above, in terms of the run-time CMS 10 of the present*

embodiment" (see col. 12:36) and "*At runtime, the interface declaration 66 is embodied*

in an interface object 52C for the interface 56..." (see at col. 14:10-13). In other words,

Bentley teaches code is generated dynamically (or at runtime). Bentley further teaches

"*Preferably, the CMS 10 of the present embodiment supports the automatic reading*

and writing of all data types by the object/persistence manager 30 of the CMS kernel

12" (see at least col. 42:27-29). This also indicates that the run-time CMS 10 supports

the automatic and dynamic generation of code.

Claim Rejections - 35 USC § 102

5.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that

form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

6.   Claim 1, 12, 16, 17, 26, and 27 are rejected under 35 U.S.C. 102(b) as being

anticipated by Bentley et al. (United States Patent No.: 5,815,415).

As per claim 1:

Bentley teaches:

> *creating a class file container object that stores source code describing a*
>
> *class* (see at least Figures 5-7; also see at least col. 12:51-53 "**In the**

programming language 53, a class declaration defines a class 54..."; col.

13:30-38 "class StormDoor : Door..."; col. 14:64-67 "A template 58 provides a

flexible programming language technique for declaring classes 54...");

adding a first source code defining a method to the class stored in the

class file container object (see at least Figures 5-7; also see at least col. 12:42-

44 "a class 54 defines member variables 60 (i.e. types of data) common to

each object 52 and methods 62 (i.e. functions) that operate on the member

variables 60"; also see col. 13:1-5 "//Declare methods..."), wherein creating a

class file container object includes selecting a class name and a super class for

the class (see at least col. 13:16-25 "Classes 54 may be derived from other

classes 54...For example, a "StormDoor" class may be derived from the

Door class...In such case, Door is a "base class" (i.e. super class or parent

class)...");

adding a second source code into the method in the class stored in the

class file container object (see at least Figures 5-7; also see at least col. 12:42-

44 "a class 54 defines member variables 60 (i.e. types of data) common to

each object 52 and methods 62 (i.e. functions) that operate on the member

variables 60"; also see col. 13:1-5 "//Declare methods...");

repeating instructions b and c to populate the class stored in the class file

container object (see at least Figure 6 – There are more than one methods

have been declared for the new class 54);

generating a tree of statement and expressions based on the class stored

in the class file container object (see at least Figures 25-26);

using the tree of statements and expressions to generate byte code for the

class (see at least col. 31:44 "**the source code is compiled into run-time code**

that may be executed on any platform 19 having an appropriately

configured run-time virtual machine"; col. 34:49-50 "**each program 96 is**

compiled into an efficient byte code format such as p-code (i.e. pseudo-

code)"); and

instantiating an instance of the new class file object (see at least col.

12:37-40 "**a class 54 is part of a schema 50 and defines an object 52 that**

may be instantiated from the class 54...");

wherein the computer program product can generate code for any type of

JAVA program ("**The dynamic framework executes on the platform and**

interfaces to the kernel, provides a platform-independent visual interface

between the CMS and a CMS user, and employs the services of the kernel"

see at least col. 6:8-12).

As per claim 12:

Bentley further teaches:

repeatedly adding a method to the class stored in the class file container

object for teach method associated with a stub generated for a remote object

(see at least Figures 6 – **There are more than one methods have been**

declared for the new class 54. Therefore, repeatedly adding methods to the

new class is inherent in order to fulfill the purpose of declaration of new

class).

As per claim 16:

Bentley further teaches:

wherein the tree of statements and expressions represents at least one

method, the at least one method comprising at least one of: a code statement, an

expression, a variable and a programming construct (see at least Figures 25-26).

As per claim 17:

Bentley further teaches:

wherein the tree of statements and expressions forms a known structure

or interface when the class is a known type (see at least col. 12:62 "**Door**"; col.

14:66 "**Shell**").

As per claim 26:

Bentley further teaches:

wherein dynamically generated code exists for the life of a server it

resides upon (**it is inherent in order to fulfill the purpose of runtime class**

generation).

As per claim 27:

Bentley further teaches:

generating executable code from the byte code by using a class loader

(see at least col. 38:25-26 "**a loader must convert the data to the format**

required for the platform 19 where the program 96 will run").

Claim Rejections - 35 USC § 103

7.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8.   Claims10 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable

over Bentley et al. (United States Patent No.: 5,815,415), in view of Click, Jr. et al.

(United States patent No.: US 6,381,737).

As per claim 10:

Bentley does not explicitly teach:

wherein the computer code implements an adaptor class.

However, Click, Jr. teaches:

wherein the program code implements an adaptor class (see at least col.

3:20-21 "**The computer code also generates an adapter when requested by**

the runtime system in order to execute the bytecode").

Therefore, it would have been obvious to one having an ordinary skill in the art at the time the invention was made to modify Bentley's approach to dynamically build an adapter class as taught be Click, Jr. One would have been motivated to generate an adapter class as taught by Click, Jr. to execute the generated byte code.

As per claim 18:

Bentley in combination with Click, Jr. teaches all the limitations of the base claim, furthermore, Bentley in combination with Click, Jr. further teaches:

> *wherein the tree of statements and expressions forms a known structure when the class is at least one of an adapter and a proxy type* (see Click, Jr. at least col. 3:20 "**an adapter**").

9.   Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over Bentley et al. (United States Patent No.: 5,815,415), in view of Glynias et al. (United States Patent No.: 6,125,383).

As per claim 11:

Bentley does not explicitly teach:

> *wherein the computer code implements a proxy class.*

However, Glynias teaches:

wherein the computer code implements an adaptor class (see at least col.

15:35-36 "**The client 110 would invoke a Java applet to create a proxy**

object, called "DNA Sequence ADH_Human").

Therefore, it would have been obvious to one having an ordinary skill in the art at

the time the invention was made to modify Bentley's approach to dynamically create a

proxy class as taught by Glynias. One would have been motivated to modify because

proxy class can be used to create another object (see Glynias at least col. 15:40-45).

10.   Claim 13 is rejected under 35 U.S.C. 103(a) as being unpatentable over Bentley

et al. (United States Patent No.: 5,815,415).

<u>As per claim 13:</u>

Bentley does not explicitly teach:

wherein the computer code for repeatedly adding a method to the class

stored in the class file container object for each method associated with a stub

generated for a remote object includes program code for:

determining a number of methods associated with the stub in a

remote interface.

However, official notice is taken that determine the number of methods is well

known to the art at the time the invention was made and modify Bentley's approach to

count the number of method declared for the class by using a counter variable. One

would have been motivated to modify because using a counter variable to count the

number of methods added to the class provides an indication when it reaches the end of

the adding iteration loop.

11.   Claim 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over Bentley

et al. (United States Patent No.: 5,815,415), in view of Whitehead et al. (United States

Patent No.: 6,085,030).

As per claim 25:

Bentley does not explicitly teach:

wherein the dynamically generated code is used for remote method

invocation skeletons, remote method invocation stubs, wrappers for database

connections, and proxies used to enforce call-by-value semantics.

However, Whitehead teaches the use of RMI, JDBC, and proxies "…**RMI, i.e.,**

proxy/skeleton interaction" (see col. 7:29) and "**ODBC/JDBC compliant databases**"

(see col. 7:67).

Therefore, it would have been obvious to one having an ordinary skill in the art at

the time the invention was made to modify Bentley's approach to use RMI, JDBC, and

proxy for remote communication with other JAVA objects and intercepting calls between

JAVA objects. One would have been motivated to modify in order to improve

convenience, compatibility or security.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Phillip H. Nguyen whose telephone number is (571) 270-1070. The examiner can normally be reached on Monday - Thursday 10:00 AM - 3:00 PM EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Y. Zhen can be reached on (571) 272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

PN
8/19/2008

/Wei Y Zhen/
Supervisory Patent Examiner, Art Unit 2191